

ITOS Makefiles

Copyright 1999-2006, United States Government as represented by the Administrator of the National Aeronautics and Space Administration. No copyright is claimed in the United States under Title 17, U.S. Code.

This software and documentation are controlled exports and may only be released to U.S. Citizens and appropriate Permanent Residents in the United States. If you have any questions with respect to this constraint contact the GSFC center export administrator, <Thomas.R.Weisz@nasa.gov>.

This product contains software from the Integrated Test and Operations System (ITOS), a satellite ground data system developed at the Goddard Space Flight Center in Greenbelt MD. See <<http://itos.gsfc.nasa.gov/>> or e-mail <itos@itos.gsfc.nasa.gov> for additional information.

You may use this software for any purpose provided you agree to the following terms and conditions:

1. Redistributions of source code must retain the above copyright notice and this list of conditions.
2. Redistributions in binary form must reproduce the above copyright notice and this list of conditions in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
This product contains software from the Integrated Test and Operations System (ITOS), a satellite ground data system developed at the Goddard Space Flight Center in Greenbelt MD.

This software is provided ‘‘as is’’ without any warranty of any kind, either express, implied, or statutory, including, but not limited to, any warranty that the software will conform to specification, any implied warranties of merchantability, fitness for a particular purpose, and freedom from infringement and any warranty that the documentation will conform to their program or will be error free.

In no event shall NASA be liable for any damages, including, but not limited to, direct, indirect, special or consequential damages, arising out of, resulting from, or in any way connected with this software, whether or not based upon warranty, contract, tort, or otherwise, whether or not injury was sustained by persons or property or otherwise, and whether or not loss was sustained from or arose out of the results of, or use of, their software or services provided hereunder.

ITOS makefiles

This document explains the conventions used in ITOS makefiles. ITOS makefiles are written for GNU (<http://www.gnu.org>) make version 3.73 GNU Make (<http://sunland.gsfc.nasa.gov/info/make/Top.html>) or higher and will not work with other make programs. In addition to GNU make, ITOS requires Perl 5 (<http://www.perl.org>) and GNU autoconf version 2.13 Autoconf (<http://sunland.gsfc.nasa.gov/info/autoconf/Top.html>) or higher.

ITOS makefiles assume the source code directories are rooted at `'$(ITOSDIR)/src'` and install programs in `'$(ITOSDIR)/bin'`, libraries in `'$(ITOSDIR)/lib'`, etc. The ITOS makefiles are named `'Makefile'`.

ITOS makefiles are intended to be portable (since GNU make itself is portable) and easy to use.

1 Makefile examples

Here's a simple *Makefile* from a program directory:

```
# $Id: makefiles.texi,v 3.8 2006/03/21 16:07:15 bgoldman Exp $
THISDIR := src/cmd/fop

BINTARGETS = fop
fop-src = fop.c

include ../Make.common
```

From a library directory:

```
# $Id: makefiles.texi,v 3.8 2006/03/21 16:07:15 bgoldman Exp $
THISDIR := src/lib/Sx

LIBTARGETS = Sx
libSx-src = SxInit.c SxObj.c SxObjDump.c SxEval.c SxFunc.c

TEXTARGETS = Sx.texi

include ../../Make.common
```

And from a parent directory:

```
# $Id: makefiles.texi,v 3.8 2006/03/21 16:07:15 bgoldman Exp $
THISDIR := src/cmd

SUBDIRS = lib cmd_dump cmd_load fop ftcpTransmit

include ../../Make.common
```

2 Commands: “make all”, “make install”, etc.

2.1 make, make all, make nall

(“make all” acts in the current directory and all subdirectories; “make nall” only acts in the current directory.)

“make” (with no explicit target) is the same as “make all”.

“make all” builds the executables named in *BINTARGETS*, the libraries named in *LIBTARGETS*, and the symbolic links named in *SYMLINKS*.

2.2 make install, make ninstall

(“make install” acts in the current directory and all subdirectories; “make ninstall” only acts in the current directory.)

“make install” builds the same things built by “make all”. In addition, “make install”:

- installs each executable named in *BINTARGETS* in *BINDIR*.
- installs each library named in *LIBTARGETS* in *LIBDIR*.
- installs each document named in *TEXTITARGETS* in *DOCDIR*.
- installs each man page named in *MANPAGES* under *MANDIR*.
- installs each Xdefault file named in *XDEFTARGETS* in *XDEFDIR*.

2.3 make clean, make nclean

(“make clean” acts in the current directory and all subdirectories; “make nclean” only acts in the current directory.)

“make clean” deletes all files which were created by “make all” and which are not named in *KEEP*.

To see what “make clean” would delete (without actually deleting anything) try:

```
make clean RM='@echo would-rm'
```

2.4 make html, make nhtml

(“make html” acts in the current directory and all subdirectories; “make nhtml” only acts in the current directory.)

“make html” generates hypertext “.html” files from the texinfo files named in *TEXTITARGETS* and places these “.html” files in a subdirectory under the directory named in *HTMLDIR*. For example, the hypertext files generated from ‘makefiles.texi’ go in directory ‘\$(HTMLDIR)/makefiles’.

Note that neither “make all” nor “make install” generate “.html” files; the only way to do this is via “make html”.

2.5 make ps, make nps, make dvi, make ndvi

(“make ps” and “make dvi” act in the current directory and all subdirectories; “make nps” and “make ndvi” only act on the current directory.)

“make dvi” generates a “.dvi” file for each texinfo file named in *TEXITARGETS*.

“make ps” generates both a “.dvi” file and a postscript (“.ps”) file for each texinfo file named in *TEXITARGETS*.

Note that “make all” generates neither “.dvi” nor “.ps” files.

2.6 making a specific library or program

Sometimes you want to do something none of the above targets handle; for example, in ‘src/olstol’ you might want to only make target `stubstol` only – not `stubstol`, `olstol`, and `syncheck`.

You can’t just say “make `stubstol`”; instead, do

```
make nall BINTARGETS=stubstol
```

3 Variables: BINTARGETS, LIBTARGETS, etc.

3.0.1 BINDIR, DOCDIR, LIBDIR, MANDIR, TCLDIR, TCWDIR, and XDEFDIR

These variables should not be set in makefiles.

- BINDIR** Defaults to $\$(TCWDIR)/bin$. This is the directory where “make install” installs executables.
- DOCDIR** Defaults to $\$(TCWDIR)/doc$. This is the directory where “make install” installs postscript documentation.
- LIBDIR** Defaults to $\$(TCWDIR)/lib$. This is the directory where “make install” installs libraries.
- MANDIR** Defaults to $\$(TCWDIR)/man$. This is the directory under which “make install” installs man pages.
- TCLDIR** Defaults to $\$(TCWDIR)/tcl$. ??? Why do we need this directory? ???
- TCWDIR** Derived from *PWD*, which is presumed to be set in the environment, and *THISDIR*, which is required to be set in the Makefile. If *PWD* is ‘/home/tsingle/tcw/src/olstol’ and *THISDIR* is ‘src/olstol’ then *TCWDIR* will get set to ‘/home/tsingle/tcw’
- XDEFDIR**
Defaults to $\$(TCWDIR)/Xdefaults$. This is the directory where “make install” installs Xdefault files.

3.0.2 BINTARGETS, ???-src, ???-lib

BINTARGETS lists executable targets to be created via “make all” and installed in *BINDIR* via “make install”.

Each bintarg in *BINTARGETS* may have additional variables, *bintarg-src* and *bintarg-lib*, specific to that target. For example, if the bintarg is *olstol*, the additional variables are named *olstol-src* and *olstol-lib*. These additional variables name the source files and libraries needed to create the executable.

bintarg-src

There may be a *bintarg-src* for each bintarg in *BINTARGETS*. *bintarg-src* lists the source files needed to create bintarg. These may be any mix of “.c”, “.1”, “.y”, or “.o” files.

bintarg-lib

There may be a *bintarg-lib* for each bintarg in *BINTARGETS*, *bintarg-lib* lists the library needed to link bintarg. For convenience, *LLIBS* and *XLIBS* may be used in the values for *bintarg-lib*.

3.0.3 HTMLDIR

HTMLDIR names the directory under which “make html” places generated hypertext “.html” files.

HTMLDIR should not be set in the Makefile; instead, it should either be set in the environment or on command line. For example:

```
make html HTMLDIR=/home/tcw/public_html
```

3.0.4 KEEP

KEEP lists files that should not be deleted via “make clean”. Most makefiles will not need to set *KEEP*, since “make clean” does a pretty good job of figuring out what to keep and what to delete.

To determine if *KEEP* needs to be set for a particular directory, examine the output from

```
make nclean RM='@echo would-rm'
```

3.0.5 LIBTARGETS, LIBATARGETS, LIBSOTARGETS, lib???-src, etc.

LIBTARGETS lists names of library targets to be created via “make all” and installed in *LIBDIR* via “make install”. *LIBTARGETS* lists names for which shared (“.so”) libraries should be created; *LIBATARGETS* lists names for a static library should be created.

Each of these names may have associated variables:

lib<name>-src

There may be a *libname-lib* for each name in *LIBTARGETS*, *LIBATARGETS*, or *LIBSOTARGETS*. *libname-lib* lists the source files needed to create the libraries. These may be any mix of “.c”, “.l”, “.y”, or “.o” files.

lib<name>-version

There may be a *libname-version* for each name in *LIBTARGETS*, or *LIBSOTARGETS* (*libname-version* doesn’t make sense for names in *LIBATARGETS*). *libname-version* identifies the version number to use when creating the shared library.

libname-version is not normally specified; when not specified, the default version number in *DEFAULT_SO_VERSION* is used.

lib<name>-sa

There may be a *libname-sa* for each name in *LIBTARGETS*, or *LIBSOTARGETS* (*libname-sa* doesn’t make sense for names in *LIBATARGETS*). *libname-sa* lists the source files needed to create the “.sa” library. These may be any mix of “.c”, “.l”, “.y”, or “.o” files.

Most shared libraries don’t have corresponding “.sa” files, therefor *libname-sa* is infrequently used.

3.0.6 LLIBS and XLIBS

These are convenience macros intended to be used with *bintarg-lib*.

LLIBS is “-ltcw” and whatever system libraries are required when linking with -ltcw.

XLIBS is the set of system libraries needed to link a motif application.

3.0.7 MANPAGES

MANPAGES lists names of man pages to be installed under *MANDIR* via “make install”.

Each name in *MANPAGES* is of the form “manpage.section” and gets installed as “\$*MANDIR*’/man’section/manpage.section”.

3.0.8 SYMLINKS

SYMLINKS lists names of symbolic links to get created via “make all”. Typically, these symbolic links are for “.h” files.

Make can’t easily determine how to create these symbolic links. Thus it is necessary for the Makefile to explicitly state how to create each link.

From ‘src/include/Makefile’:

```
SYMLINKS = dsp Sx.h Log.h LogP.h Proc.h ProcP.h
```

```
...
```

```
dsp:
    ln -s ../dsp/include dsp
```

```
Sx.h:
    ln -s ../lib/Sx/Sx.h .
```

```
Log.h LogP.h Proc.h ProcP.h:
    ln -s ../lib/xtcw/$ .
```

3.0.9 TEXTARGETS

TEXTARGETS lists texinfo files from which hypertext (“.html”) and postscript (“.ps”) documentation is generated.

3.0.10 THISDIR

THISDIR is required to be set at the top of each ‘Makefile’. *THISDIR* shows where the directory is relative to the root of the src tree. For example, the ‘Makefile’ for *olstol* (‘src/olstol/Makefile’) sets:

```
THISDIR := src/olstol
```

3.0.11 XDEFTARGETS

XDEFTARGETS list names of Xdefault files to be installed in *XDEFDIR* via “make install”.

Table of Contents

ITOS makefiles	1
1 Makefile examples	2
2 Commands: “make all”, “make install”, etc.	3
2.1 make, make all, make nall	3
2.2 make install, make ninstall	3
2.3 make clean, make nclean	3
2.4 make html, make nhtml	3
2.5 make ps, make nps, make dvi, make ndvi	4
2.6 making a specific library or program	4
3 Variables: BINTARGETS, LIBTARGETS, etc.	5
3.0.1 BINDIR, DOCDIR, LIBDIR, MANDIR, TCLDIR, TCWDIR, and XDEFDIR	5
3.0.2 BINTARGETS, ???-src, ???-lib	5
bintarg-src	5
bintarg-lib	5
3.0.3 HTMLDIR	6
3.0.4 KEEP	6
3.0.5 LIBTARGETS, LIBATARGETS, LIBSOTARGETS, lib???-src, etc.	6
lib<name>-src	6
lib<name>-version	6
lib<name>-sa	6
3.0.6 LLIBS and XLIBS	7
3.0.7 MANPAGES	7
3.0.8 SYMLINKS	7
3.0.9 TEXITARGETS	7
3.0.10 THISDIR	7
3.0.11 XDEFTARGETS	8